MULTIMEDIA UNIVERSITY

STUDENT ID NO

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 1, 2016/2017

TSE3151/TSD2711 – SOFTWARE DESIGN

(All Sections/Groups)

18 OCTOBER 2016
2:30 PM – 4:30 PM
(2 Hours)

INSTRUCTIONS TO STUDENTS

1. The total number of pages for this examination paper is nine (9) pages including the cover page.

2. This examination paper contains a total of six (6) questions divided into two (2) different sections, Section A and Section B, respectively. Each section contains three(3) questions.

3. You are required to answer a total of four (4) questions. You must answer any two(2) questions from Section A, and any two (2) questions from Section B.

4. Each question carries an equal score of 20 points. The total score for this examination paper is 80 points.

SECTION A

(Answer any two(2) questions in this section)

QUESTION A1

A1(a) Answer the following questions. You may provide examples in your answers.

(a1) Discuss four (4) reasons why a good software designer must be a person with knowledge, skills, relevant experience and professionalism in many different aspects in software engineering. [4 points]

(a2) Describe three (3) impacts for not having sufficient understanding of the domain knowledge with regards to the requirements of a particular software system. [3 points]

(a3) Provide three (3) examples of behavioral requirements for the control software of a Domestic Washing Machine (DWM).

[3 points]

A1(b) Discuss the following issues in terms of what they are and the context of how each term affects software design. You may provide appropriate examples in your answers, not exceeding 5 sentences per answer.

(b1) Unambiguous requirements.
(b2) Consistent requirements.
(b3) Non-Functional requirements.
(b4) Requirement constraints.
(b5) Incomplete requirements.

[5 x 2 points = 10 points]

Continued ..

QUESTION A2

A2(a) Answer the following questions. You may provide examples in your answers.

(a1) Explain three (3) ways how data and class elements are related to use-cases and use-case scenarios. [3 points]

(a2) Why is it *not correct* to say that encapsulation in object-oriented design is about information hiding? [3 points]

(a3) List any four (4) of the five (5) SOLID principles in object-oriented design. [4 points]

A2(b) Answer the following questions. You may provide examples in your answers.

(b1) Draw a sequence diagram for the following scenario:

Facebook uses the OAuth protocol framework which enables web application (called "client"), which is usually not the FB resource owner but is acting on the FB user's behalf, to request access to resources controlled by the FB user and hosted by the FB server. Instead of using the FB user credentials to access protected resources, the web application obtains an access token.

Web application should be registered by Facebook to have an application ID (client_id) and secret (client_secret). When request to some protected Facebook resources is received, web browser ("user agent") is redirected to Facebook's authorization server with application ID and the URI the user should be redirected back to after the authorization process.

User receives back Request for Permission form. If the user authorizes the application to get his/her data, Facebook authorization server redirects back to the URI that was specified before together with authorization code ("verification string"). The authorization code can be exchanged by web application for an OAuth access token.

If web application obtains the access token for a FB user, it can perform authorized requests on behalf of that FB user by including the access token in the Facebook Graph API requests. If the user did not authorize web application, Facebook issues redirect request to the URI specified before, and adds the error_reason parameter to notify the web application that authorization request was denied.

[6 points]

Continued ..

(b2) Draw a use case diagram for the following scenario:

An **Online Public Access Catalog (OPAC)** is e-Library website which is part of **Integrated Library System** (ILS), also known as a **Library Management System** (LMS), and managed by a library or group of libraries. Patrons of the library can search library catalog online to locate various resources - books, periodicals, audio and visual materials, or other items under control of the library. Patrons may reserve or renew item, provide feedback, and manage their account.

[4 points]

Continued ..

QUESTION A3

A3(a) In software design, *Actors* and *Use-cases* help designers identify the required behaviors and functions in a software system. Actors need not necessarily be persons interacting with the software. Actors can also be software subsystems or software modules that react to software events or that use the software functions. To help identify system *Actors* and *Use-cases,* it is common practice that designers use prompt questions.

List down the possible *Actors* and *Use-cases* based on your answers to each of the prompt questions below.

(a1) Who internally manages software users and software security? [4 points]

(a2) What happens if the software system fails? [3 points]

(a3) How can the performance of the system degrade? [3 points]

(Note that using prompt questions is practiced widely by police in order to get facts and information regarding cases in their investigations. This is what this question is all about i.e. to get the Actors and Use-cases based on answers to the prompt questions).

A3(b) Consider the following five (5) statements. Explain what each statement means. *(Note that by explain, we mean you have to explain further by making the statement clear and free from ambiguity).*

(b1) Object serialization and de-serialization are about abstract data transfer.
(b2) By inheritance, the derived classes reused all the base class codes.
(b3) By polymorphism, classes can be designed as abstract.
(b4) Object-oriented design implements nonlinear logic and not serial logic.
(b5) Structured design emphasizes on procedures while object-oriented design emphasizes on real world models.

[5 x 2 points = 10 points]

Continued ..

SECTION B

(Answer any two (2) questions in this section)

QUESTION B1

B1(a) In this question, we consider software design architectures. You may provide examples in your answers.

(a1) A monolithic application implements a single-tiered software design. Provide two (2) advantages of this design architecture. [3 points]

(a2) The design of *Facebook* and *Twitter* applications is considered as Distributed and Concurrent architecture. Describe how these applications operate. [4 points]

(a3) The CAMSYS system at Multimedia University is possibly designed based on the combined N-Tiered and Data-Centric architectures. Clarify what these architectures mean.

[3 points]

B1(b) Consider the following five (5) statements. Explain what each statement means. You may provide examples in your answers.

(b1) In object-oriented design, computer programs are created by making them objects that interact with one another.

(b2) Compiled applications are targeted towards a particular platform and so are platform dependent.

(b3) One of the advantages of the Interpreter design architecture is that it makes the application platform independent.

(b4) It is not true that almost all hardware manufacturers provide software drivers for their hardware in order to make them platform independent.

(b5) A transaction server is designed to handle thousands of simultaneous users.

[5 x 2 points = 10 points]

Continued ..

QUESTION B2

B2(a) Answer the following questions. You may provide examples in your answers.

(a1) Describe the time-sharing and multi-tasking model in software design. Is this design only applicable to operating systems and not software applications? [4 points]

(a2) Explain the difference between multi-threading and multi-processing designs in terms of their memory environments. [3 points]

(a3) How does semaphore work in controlling access to shared resources in parallel computing designs? [3 points]

B2(b) Answer the following questions. You may provide examples in your answers.

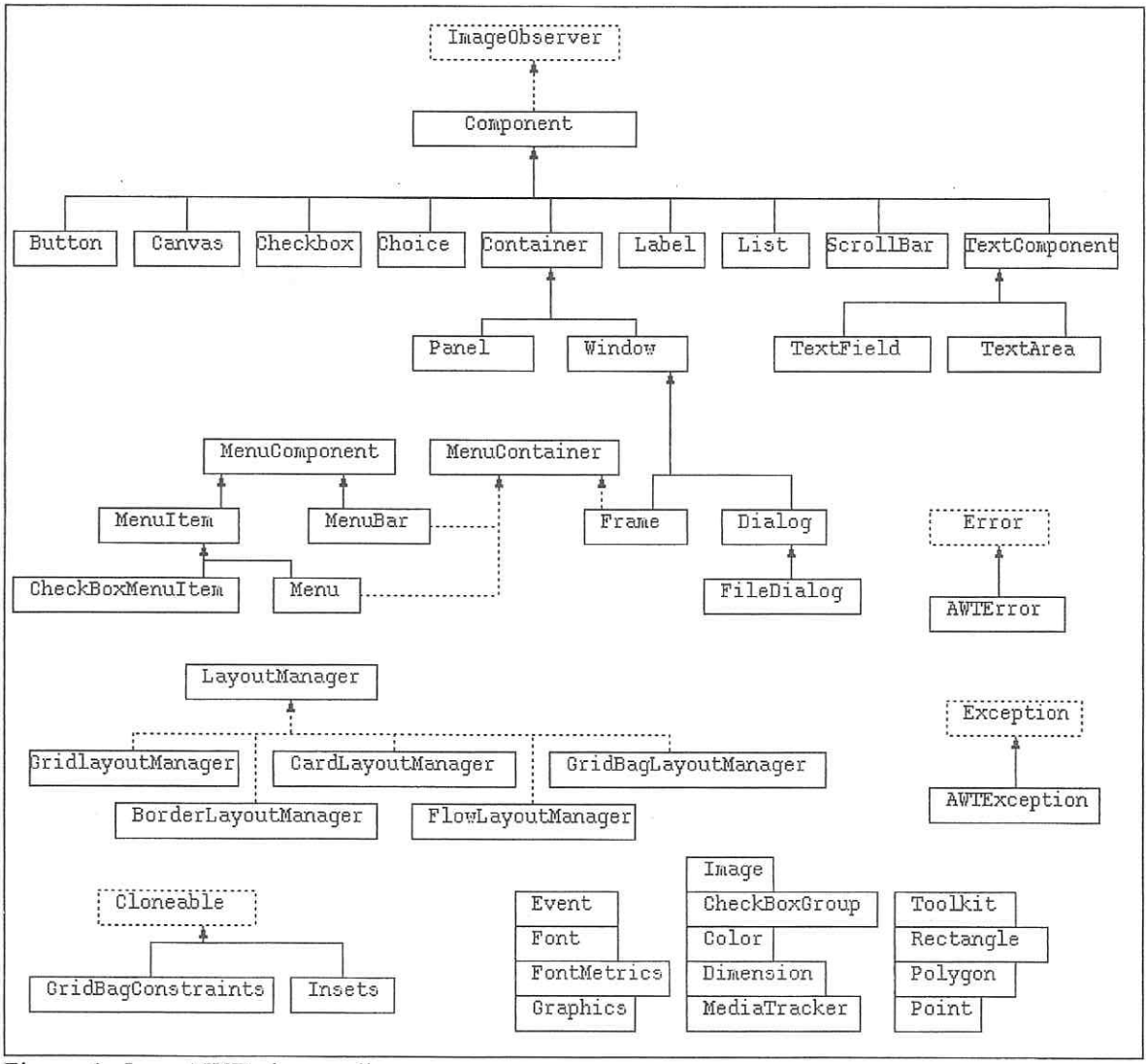

Figure 1: Java AWT classes diagram

Continued ..

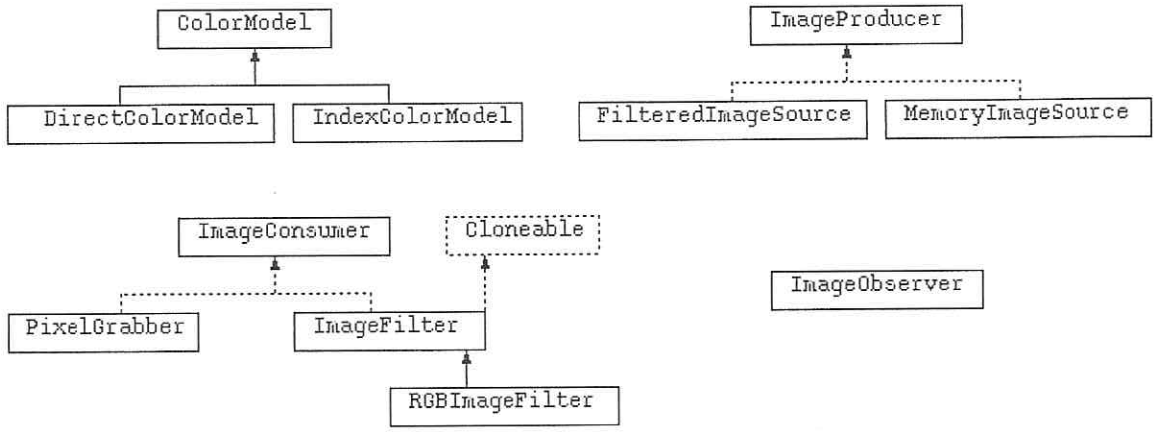

Figure 2: Java Awt.Image classes diagram

Refer to Figure 1 and Figure 2. The figures shows the class diagram for the Java AWT and Java Image respectively. Solid arrows to represent inheritance and dashed arrows to represent (interface) implementation. A dashed box around a class means that it is part of another Java class library.

Identify and explain the usage of the following Design Patterns in these the Java AWT and Java AWT.Image class hierachies. For each design pattern, draw a class diagram to show the relevant classes in each of the design patterns.

1. Observer
2. Strategy
3. Prototype
4. Factory

[4 x 2.5 = 10 points]

Continued ..

QUESTION B3

B3(a) Answer the following questions. You may provide examples in your answers.

(a1) Explain four (4) advantages of using existing software frameworks compared to designing software applications from scratch. [4 points]

(a2) Software application designs can influence our choice of programming language. On the opposite, our choice of programming language can influence our software application design. Which of these two statements is correct? Is this situation a chicken-and-egg issue? [3 points]

(a3) Clarify the three (3) differences between a recursive algorithm and an iterative algorithm. In what situations do we implement one algorithm over the other?

[3 points]

B3(b) Answer the following questions. You may provide examples in your answers.

(b1) Provide two (2) advantages of using Cascading Style Sheets (CSS) in the design of web applications?

(b2) Describe three (3) major challenges in the processing of Big Data.

(b3) Give two (2) reasons why is it that many organizations are still skeptical and hesitant about hosting their business applications and data in the cloud infrastructure?

(b4) What does the term Internet-of-Things (IoT) mean?

(b5) Describe three (3) market segments where IoT applications contribute significant social and economic benefits to mankind.

[5 x 2 points = 10 points]

End of Page